SF 298 MASTER COPY KEEP THIS COPY FOR REPRODUCTION PURPOSES

REPORT DOCUMENTATION PAGE

Form Approved OMB NO. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
1995

3. REPORT TYPE AND DATES COVERED
Technical Report

4. TITLE AND SUBTITLE
An Approach to Building a Highly Parallel Computer System.

5. FUNDING NUMBERS
DAAH04-94-G-0024

6. AUTHOR(S)
O.Zhukov, C. Ordonez, N.Rishe

7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(ES)
Florida International University
School of Computer Science
University Park Campus
Miami, FL 33199

8. PERFORMING ORGANIZATION REPORT NUMBER
FIU SCS Technical Report
#95-1

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Office
P.O. Box 12211
Research Triangle Park, NC 27709-2211

10. SPONSORING / MONITORING AGENCY REPORT NUMBER
ARO 32427.5-MA-SDI

11. SUPPLEMENTARY NOTES
The views, opinions and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy or decision, unless so designated by other documentation.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution unlimited.

12 b. DISTRIBUTION CODE

19960522 016

13. ABSTRACT *(Maximum 200 words)*

Today's commercial prospects of massively paralled computers p[illegible] performance at the level of hundreds of gigaflops. Perhaps computers with performance at teraflop levels can become available in the near future. However, already today's leading researchers are considering the following question: What kinds of architecture and technology must be used for creating supercomputers in order to make them commercially viable and suitable for the industry? Providing performance up to petaflops is quite a new problem. Analyzing achievements in optical techniques has led us to believe that there are possibilities of projecting an optoelectronic or even a purely optical, highly paralled computer which might get us there.

In this article we sconsider the problems regarding a processor interconnection network using distributed shared memory, propose a possible solution to these problems and examine a concept for a high performance and highly reliable processor, based on optics. We will also demonstrate the important role which the residual number system should play in building such a processor and the benefits of using nomography within an optical processor to improve performance.

14. SUBJECT TERMS

15. NUMBER IF PAGES

16. PRICE CODE

17. SECURITY CLASSIFICATION OR REPORT
UNCLASSIFIED

18. SECURITY CLASSIFICATION OF THIS PAGE
UNCLASSIFIED

19. SECURITY CLASSIFICATION OF ABSTRACT
UNCLASSIFIED

20. LIMITATION OF ABSTRACT
UL

DTIC QUALITY INSPECTED 1

An Approach to Building a Highly Parallel Computer System.

FIU SCS Technical Report 95-3

N. Rishe, O. Zhukov, C. Ordonez

High-performance Database Research Center
School of Computer Science
Florida International University
University Park, Miami, FL 33199
Telephone: (305) 348-2025, 348-2744
FAX: (305)-348-3549; E-mail: rishen@fiu.edu

This research was supported in part by NASA (under grant NAGW-4080), ARO (under BMDO grant DAAH04-0024), NATO (under grant HTECH.LG-931449), NSF (under grant CDA-9313624 for CATE Lab), and State of Florida.

Abstract

Today's commercial prospects of massively parallel computers peak in performance at the level of hundreds of gigaflops. Perhaps computers with performance at the teraflop levels can become available in the near future. Researchers are considering the following question: What kinds of architecture and technology must be used for creating supercomputers in order to make them commercially viable and suitable for the industry? Providing performance up to petaflops is quite a new problem. Analyzing the achievements in optical techniques has led the authors to believe that there are possibilities of designing an optoelectronic, or even a purely optical, highly parallel computer which might get us there.

In this paper we consider the problems concerning a processor interconnection network using distributed shared memory, propose a possible solution to these problems, and examine the concept of an optic-based high-performance highly-reliable processor. We also demonstrate the important role which the residual number system (RNS) should play in building such a processor and the benefits of using nomography within an optical processor to improve performance.

1. Introduction.

Since the early days of computers the most powerful machines were used for massive scientific computations. During the last few years the need for supercomputers has increased extraordinarily in the field of scientific research. Today the use of high-performance computers has become broadly known as the third approach to scientific studies.

Traditionally, science was based upon experimental and theoretical approaches. Supercomputers in science have become a strategic resource for performing advanced research and developing high technology. Many applications concerning global problems of human welfare and problems at exciting frontiers of science can only be solved by supercomputers. For example, the goal of improving atmospheric modeling resolution to a 5-km scale and providing timely results is believed to require up to 20 teraflops of computational power. Some other problems in the nuclear and astro-physics arena demand similar performance. Furthermore, it is believed that digital signal processing will require performance to the P-flops (petaflops or 1000 teraflops) level. Thus, today's challenging applications require computer speeds in orders of magnitude greater than the gigaflops available and the teraflops that may be achieved in the near future.

As the computer technology has evolved from transistors to integrated circuits, to very large scale integration (VLSI), the physical dimensions of logic has decreased, therefore the restriction due to the engineering and control of flip flops has been reduced substantially. In these transitions, reducing propagation delays further increased computing speed by increasing switching speed and circuit density. Such breakthroughs are less anticipated in the future because circuit density is fast approaching the restrictions due to engineering and control of digital logic designs. Furthermore, in the past the speed of processors increased by a factor of ten times every seven years but the speed of light and quantum mechanics laws determine the physical limits of this process. Attempts to create faster processors, buses and memories have led to the use of very complex logic and expensive technologies.

Currently, super-computing almost always refers to vector and/or parallel computers. While all of the components of supercomputers may help run your programs faster, it is the vector and parallel processing that provides the greatest speedup. In a vector computer there is integrated processing of an array of numbers. The hardware allows for a sequence of identical operations to be performed on data arranged in an array. The operations are to be executed simultaneously on all N elements of the array. In a parallel computer there is simultaneous and independent processing of different programs or different segments of the same program, in different processors. It seems inevitable that parallel processing will attain speeds in the range of 1,000T - 10,000T flops. Possibly this explains why the computer science community and the industry appears to have such high interest in massively parallel computers. The modern supercomputers in service today are considered massively parallel processor (MPP) machines.

2. Problem Description

Recently, many new parallel computer architectures have been emerging into the market, all aiming to increase performance. Today's MPPs peak in performance at hundreds of gigaflops. Perhaps, if we fully utilize existing, proven technologies, we could increase performance to a few terraflops. These "teraflop computers" could become available in the near future. Improving performance significantly beyond teraflops will demand essential innovations in hardware architecture, in packing, and in device technology.

If we want to make broad usage of parallel architectures, it is important that they perform a wide variety of applications, without excessive programming difficulty. To maximize both high performance and wide applicability, parallel architectures should provide:

(a) the ability to support hundreds to thousands of processors

(b) high-performance individual processors

(c) a single shared address space

(d) a suitable relationship between cost and performance

In designing parallel computer systems, three fundamental questions [41] come to mind:

(1) What should be the nature, size, and number of processors?

(2) What should be the nature, size, and number of memory modules?

(3) How to interconnect all the processors and memories?

In general, there is a link between these three questions. The number of processors and memory modules has some impact on the interconnection strategy used, as in a small number of powerful processors with a simple communication hardware connecting them or a large number of low performance processors with a complex communication strategy. Although there are some differences among processors and memory designs, the essential question relates to how the pieces (memories and processors) are put together.

Here it is necessary to review certain issues of parallel computer architectures.

There are at least two ways to categorize parallel systems: based on how many instructions and data streams they have; and based on the granularity of the algorithm used. We now describe each method.

Parallel machines can be divided into three subsequent categories (Flynn, 1972) based on how many instructions and data streams they have:

(1) SISD: Single Instruction stream, Single Data stream.

(2) SIMD: Single Instruction stream, Multiple Data stream.

(3) MIMD: Multiple Instruction stream, Multiple Data stream.

Within the MIMD category we see two basic processor interconnection strategies. Shared memory, where any processor can communicate with any other via a shared address space; and message passing (also called distributed memory), where there is communication between processors via predefined messages. The message passing MIMD machines refer to processor organization as processor topology, which specifies how processors are connected in a parallel computer. The shared memory approach is suitable for a wider range of problems in contrast with message passing, which is simpler but limited to those problems where the architecture is well matched to the application. In general, shared memory places more of the burden on the hardware designer, whereas message passing places it more on the programmer. We find at least two flavors of the shared memory design: tightly coupled and loosely coupled processors.

Tightly coupled processors share a global memory through the use of a central switching mechanism. This switching mechanism determines the processor organization. We can divide the switching mechanisms into three types: common bus, crossbar switch, and multi-stage switched network. The common bus is simple in concept but limited in its future potential of improvement. The bandwidth of a bus is the product of the clock frequency and the width of the data path in bits. This product has to match the total processing power of the attached processors. Bus widths over 72 bits are rare, and the only other way to increase the bandwidth of the bus is to speed up the clock frequency. But the same technological advances that would increase the bus clock rates, would also make faster processors possible, so the ratio would remain about the same, and the number of processors that can be supported on a single bus will remain limited [40]. The crossbar switch network uses multiple sets of switches to connect processors with memory elements. The problem with this network is that the switch still adds an N^2 cost component (on an equal number of processors and memory you have N^2 crosspoints). The crossbar switch is not suitable for highly parallel systems. The multistage switched network is a collection of 2x2s or slightly larger crossbar switch elements arranged in an array.

Loosely coupled processors share an address space which is obtained by combining local memories from all CPUs.

Another way to categorize a parallel machine is based on granularity where a grain is defined as the measure of the computational work to be done sequentially. There are three categories: coarse grain, medium grain, and fine grain parallelism. Coarse grain parallelism refers to separate programs running on separate computer systems, with the systems coupled via a conventional communication network. Medium grain parallel systems has several processors executing (possibly different) programs simultaneously while accessing a common memory. In this case programs have different, independent, parallel subroutines running on different processors. Fine grain parallelism tends to be used in custom-designed machines. As the granularity decreases and the number of nodes increases, the need for fast communication between nodes becomes a requirement. Communication may be via a central bus or shared memory for a small number of nodes (less than 8) or through some form of a high speed network for massively parallel machines. In this case the compiler divides the work among the nodes.

Now having reviewed the general structure of parallel computers, we can clasify some of the existing machines by the number of their processors:

1. Small number (2-8) of very powerful processors using shared memory. Here each processor usually has a vector facility attached. These machines are the traditional supercomputers. They include: the Cray 2 (4 CPUs), IBM ES/9000 (8 CPUs), and the Convex C2xx (4 CPUs). The parallel processing in this class serves to increase the throughput by working on different jobs, in different CPUs, as well as having all the processors work on the same job.

2. 8-256 medium power processors utilizing a shared memory strategy. Machines in this group include: the Alliant FX (28 CPUs), the BBN TC2000 Butterfly (256 CPUs), and the Sequent Balance (30 CPUs). They are not yet considered general purpose machines and will probably grow in number as some of the previous group's machines become bigger.

3. 100s-1000s medium power processors using individual memories. Systems in this class include: the Hypercube NCUBE (512-8194 CPUs), the Intel iSPC/2 (128 CPUs), and the Connection Machine CM-5 with 16,000 processors.

Because communication often limits the speed in these machines, some have developed intricate interconnection schemes. Most new machines are in this class. The individual processors are frequently versions of general purpose RISC-processors such as the SPARC nodes that comprise the CM-5. These are often MIMD machines employing a message passing system.

4. 1000s low power processors in a synchronous architecture. Machines in this group include the Connection Machine CM-2 (up to 65,536 CPUs) and the MasPar (16,384 CPUs). These are SIMD machines and are relatively easy to program, thanks to sophisticated SIMD compilers.

It has been argued that medium grain parallel systems are most likely to predominate in the future since they provide an amortization of processor cost. It is also notable that the high speed RISC-workstations compute at rates just recently reserved for supercomputers. As these start to be mass produced, their price should fall and then hundreds or thousands of them will be used as processors. This in turn means that the efficient use of cache and pipelines, so important to attain a high processing rates on the workstations, would also be important on these parallel machines. The problem with this type of parallelism is that the computation speed of these workstations is already hundreds of times faster than the communication speeds available. The latency due to communication limits the performance. This communication bottleneck is handled by completing more of the computation at each node before passing it to another node, implying a coarse grain system using distributed memory and efficient use of cache and pipelines at each node.

Going back to what we said before, there are two basic interconnection strategies within MIMD: shared memory and message passing. Keeping in mind these two key design issues, the shared (common) memory machines, due to their nature, typically belong to the class of course grain computers. These can be used for a wider range of applications. The message passing machines have fine granularity with fixed communication patterns, and are more suitable for a limited number of problems. An example of such architecture is the FLEX32 [1]. There are hybrid architectures, such as the BBN Butterfly [2], in which memory is distributed but communication between processors is performed via shared variables.

The importance of parallel data transport and the complexity of communications in computers has increased tremendously. This is obvious when we consider the Von Neumann computer in contrast with present parallel computers containing many tightly coupled processors[3]. Expectations are that processing and connectivity aspects will merge more and more in future computer architectures.

One of the major problems is the construction of a suitable interconnection network which could provide fast and flexible interprocessor communications at a reasonable cost. The cost-effectiveness of any arbitrary network design is governed by such factors as the number of processors, the computational tasks to be performed, the required speed of the interprocessor data transfers, the complexity of the actual hardware topology of the network, and any cost constraints on its construction. The ideal goal of any parallel architecture is to obtain a near linear relationship between performance and the number of processors. Most of these architectures rely on a restricted topology to keep the cost of the communication hardware low.

The principal features pertaining to performance criteria in a parallel architecture are : small size (volume), uniformity, extendibility, short connections(wires), efficient lay out, simple routing algorithm, fixed node degree, and fit to available technology [4-7].

The size of a system topology is also a very important parameter. This is defined as the maximum number of times that a message can be forwarded between routers when transferring from one processor to another. The number of connections is a key cost feature. Node degree, which is the maximum number of connections per node, is an important parameter for a routing algorithm. The same router can be used in a network of any size, and the network is easy to extend, if the node degree is fixed.

Any decision regarding the selection of a processor topology is a compromise between cost and performance, and it strongly depends on the intended application. For example, one of the main reasons for the hypercube to be the most widely used network is the fact that it can be simply reconfigured into some other network, such as a two- or three-dimensional mesh, a ring, or a tree network [6]. It also requires the most connections of any star topology (2^n processors requires n connections from each node). Because of the need of more connections, more hardware is required at a greater cost, but the performance improvement and the reconfigurability of the hypercube clearly outweighs the cost increase.

3. Optical Interprocessor Communications and Distributed Shared Memory.

As was expected and observed for highly parallel computers, such as in the NCube and the Connection Machine, the message or data communication becomes a bottleneck [8]. While microprocessors can be densely packed on a board, the number of interconnection lines or channels required by each processor to communicate with the others in the network will limit the size of the system which can be implemented. The data communication problem is further exacerbated as the next generation of powerful microprocessors moves from 32- to 64-bit architectures and beyond. The communication off-chip and off-board are limited to single bits because of the lack of high density interconnection technology. In a communication network which uses packet switching, the number of bits per packet will be much greater than the word size. Even if multiplexed electronic lines are used to transfer all the data from each processor on a single channel, the bandwidth required, for a 40 MHz 64-bit processor (such as the i860), is 2.6 GHz. With the current technology, this bandwidth can be provided only using state-of-the-art GaAs logic circuits. The interconnection problem becomes worse when more sophisticated processors are used. As faster processors become available, such as the RISC-processors operating at 100-400 MHz, no electronic circuit technology can provide the space bandwidth required to supply parallel data transfer from one processor to another.

Recognizing that semiconductor technology alone cannot provide the increasing demand for computing performance, there has been a marked movement among computer designers to optical and opto-electronic architectures. Thus one possible solution to the given problem is to use optical interconnections.

Extensive work has been done in the area of optical interconnection networks and their suitability to parallel computers [9-28]. Characteristics such as increased fan-out, very large bandwidth, high reliability, low power requirements, reduced crosstalk, and immunity to electro-magnetic interference make optical networks very desirable.

In general, various media, topologies, and interconnection methods can be used for building optical networks. How well interconnection density requirements can be satisfied depends on the nature of materials and the devices used for implementing the interconnections. In the process of selecting an appropriate optical technology, it is necessary to account for the interconnection density as well as for the loss and crosstalk of the optical medium, and the performance of components required to use this medium in interprocessor communications. The net effect of various perturbations on the optical signal is characterized complexly by crosstalk and loss of the system, optical power levels, and the design of the receivers [11].

The most suitable optical sources include LEDs, lasers, and various types of external modulator-fed lasers. LEDs have low efficiency and the reliability of lasers depends essentially on temperature. One possible approach to this problem includes external modulation of a laser with sufficiently high power, possibly temperature stabilized and optically isolated.

Free-space implementations of the shuffle exchange connectivity (SEN) range from the use of bulk optics to holographic elements with complex connectivity. The advantages of free space interconnections are associated with the use of three dimensions. This may imply that the use of system architectures differ from the electrically interconnected case. The most attractive modulators for realizing free-space technology are spatial light modulators [12]. An increase in interconnection density over bulk optical implementations may be obtained through the use of micro-optics via the reduced sizes of discrete elements with diameters of a fraction of a millimeter [21]. Holograms offer the possibility of implementing mapping of arbitrary regularity using one optical element per board of a multiboard system. Research has been published [23] already considering the use of holograms to implement the space interconnection topologies required in the multi-processor systems.

As an alternative to free-space architectures, we can consider guide-wave implementations of the shuffle exchange for mapping interconnection networks. In this case an apparently simple solution to the interconnection problem is found in the form of optical fibers. Developed primarily for the telecommunications industry, this technology is relatively mature. All the components required for the mappings are available commercially, albeit with a lower density than required for our application. The polyimid waveguides have better characteristics in comparison with optical fibers [26]. Using appropriate polyimides and fabrication procedures, it is possible to fabricate multi-mode optical waveguides with dimensions from a few microns to several hundreds of microns. Practical analysis showed that the most attractive solution to the interconnection system appears to be a polymer waveguide backplane system. Interconnections between logic planes may likely employ classic optical components such as mirrors, lenses, beam splitters and holographic deflectors. Much work has been done and continues to be done in these areas, and there is confidence that these results will provide the required components and subsystems[21].

As mentioned above, there are various schemes, each characterized by complexity and functionality, that may be used for building a parallel architecture. Two main types of such schemes of practical interest are: systems with limited direct connections in which interprocessor communications are defined by the interconnection topology and systems with full direct connections, in which processing nodes and network are fully integrated, thus providing direct communication on a local physical basis. The first type is characterized by a more economical connection at the expense of an increase in the delay of a signal when its propagation requires passing through more than one node. The second type is characterized by obtaining the maximum possible communication speed, but it is less economical.

Currently, topologies like the N-Cube and Shuffle Exchange Interboard Connectivity (SEN) are considered the most suitable for networks connecting a large number of processors. On the other hand, a rational design for any configuration based on a scaleable network connecting a very large number of processors implies a multistage network scheme. The SEN represents a network which is more sophisticated than a ring network but simpler than a fully connected network. The SEN is attractive in terms of optics because of the relative ease with which the perfect shuffle can be implemented[28]. The throughput of a replicated SEN can be substantially higher than that of N-Cube network.

In general, MPPs can be implemented in several different ways. For example, one way would be to use electronic processors and employ an optical interconnection network. This combination would make the design of processors simple while retaining the parallelism of a large number of interconnections. One possible way to do this would be to use, as described above, waveguide interconnections and nearly conventional electronic systems [11]. Another would be to integrate the electronic processors in a 2D-array with integrated optical transmitters and receivers[12]. Decoupling of the network and of all the processors allows significant performance improvements in an all-optical implementation.

Optical communications also result in highly fault-tolerant systems due to the high connectivity. Photonic switching can be implemented using any of the following three methods: space division multiplexing, time division multiplexing, or wavelength division multiplexing. Currently, wavelength division multiplexing results in better performance characteristics for average distance, diameter, and average packet delay in comparison with a hypercube topology. Multiple optical channels can be formed out of a single fiber by using wavelength division multiplexing to create multiple access channels. This approach is to circumvent the speed mismatch between the optics and the interface electronics: multiple channels are created on a single fiber rather than creating a single very fast channel. Wavelength selectivity can be reached either with a coherent receiver, or a tunable filter with direct detection. The first approach is more expensive, but has higher channel selectivity. A lower cost alternative is the tunable filter with direct detection which is based on electro-optic devices that have switching speeds in the range of nanoseconds.

The low loss region of a single mode optical fiber is characterized by a bandwidth of about 30 Thz. This illustrates the problem of a speed mismatch: optical media are capable of speeds far exceeding the maximum speeds of the electronic interface components. In our case each channel operates at the data-rate limited by the electronic components. This achieves a significant improvement in bandwidth utilization, allowing concurrent transmission along the multiple channels. The assessment of hybrid optical networks also reveals that optics could potentially provide the high connectivity needed for high bandwidth and high density interconnection networks in the MPPs architectures.

An interesting approach to projecting multistage interconnection networks (MIN) with fixed optical interconnections is illustrated in [30] . In the present implementation of MIN, data is transmitted in one direction. The capability of the three-dimensional physical space for providing 3 independent directions of light propagation is not fully utilized. But if we separate each switching array into receiver arrays (arrays of detecting diodes) and sender arrays (surface emitting laser arrays or modulators), and by bending the MIN, one can obtain the compact architectures in which light beams cross from many sides [30]. The major advantage of this architecture is its compactness, which is assumed to promise increased mechanical stability and the separation of optics (inside the architecture) and electronics (outside) with almost unlimited space for the latter [30].

It also is necessary to notice another important problem concerning organization of a system memory. First, an addressable memory is central to all traditional or existing computational models [29]. An integrated performance of multiple processor systems is usually limited by a memory access. Second, a fine-grained architecture, as a rule, demands high connectivity as well as high throughput of data or messages between processors. As mentioned before, there are two main types of MPS memory organization: shared memory and distributed memory (message passing systems). Significant advantages of shared memory systems are the ease of programming and good performance with small systems. A main disadvantage is the limited system size capability due to the limited communication bandwidth and the increase in memory contention. Distributed memory or message passing systems, on the other hand, have a strong advantage in their ability to support a large system size (systems which can extend into the massively parallel region) but require a mechanism involving packet formation, routing, and decoding. This complexity together with the relative programming difficulty has hampered distributed memory systems from becoming a dominant approach to parallel architecture.

In 1986, an architecture was proposed called distributed shared memory (DSM) [29]. The DSM approach incorporate both distributed and shared memory system concepts to extract the strengths of each, while balancing their respective weaknesses. DSM systems are constructed as a distributed memory systems, but appears to the programmer as a shared memory system. A variety of ways exist to simulate distributed shared memory. One way is by mapping the memory modules, located at each processor, into a global address space. This organization is not limited by the memory assigned to each processor. The access to shared memory modules involves a network access. Thus, the network latency is a part of memory latency.

Different dynamic memory allocation schemes may be proposed. A memory allocation and a process allocation are closely related in parallel computers. A dynamic allocation needs the resource requirements of a job to be known before it is assigned to a particular node, and may be used to reduce network accesses. An alternate approach is to uniformly allocate the address space to all the nodes. One way would be to assign address space to each processor with a possibility of accesses by other processors through message passing. The DSM can also be implemented as a virtual memory shared by all the processors as in where memory mapping managers maintain memory coherence. The DSM system combines the concepts of shared memory systems and distributed memory systems on the base of a fast interconnection network. The optically interconnected DSM overcomes the problems of network delays through the use of photonic communications. In general, it eliminates the need of having a centralized shared memory, and facilitates the use of distributed memory.

A result of the high communication capacity is the simplification of a global address mapping problem. There is a possibility in the optical DSM system to avoid the complexity of the above mentioned types of dynamic allocation, and to implement the simpler uniform allocation method since network latency and bandwidth are not longer principal limiting factors. In the uniform allocation scheme, when a request is sent by the processor to the shared memory, it can be directed to any one of the memory modules. With an optical network, which has a very fast transmission rate and many channels, the fixed allocation scheme results in requests being dispersed uniformly over all the memory modules in the system. This results in lesser traffic, and lower delay for a request because of reduced queue lengths at the slow memory. In general the photonic network may support a throughput rate far beyond the packet processing capability of a typical node.

It is necessary to notice the following. In the DSM system each processor is associated with a portion of the system address space, which can be accessed by the other processors through message passing. Each processor typically executes from its associated memory as in distributed systems. At the same time, all processors have access to the entire address space as with shared memory systems. The required blocks are brought into the address space allotted to the processor, and used for process execution. This configuration is very attractive in that it combines the topological advantages of distributed systems with the programming advantages of shared memory systems. The application base of the resulting system is wider than for each of the two approaches individually, moving this system in the direction of general-purpose parallel computer.

4. High Performance and High Reliable Optical Processor

There exist a problem with increasing the performance of separate processors working autonomously in a complex. Here, most of the development is directed towards accelerating separate algebraic operations by using the latest technology, like accelerating of spreading carries between separate positions of an arithmetic-logic unit(ALU). Another direction being taken is the development of an ALU containing separate functional blocks like fixed point, floating point, decimal arithmetic, multiplication, element functions, and so on. Since 1950's, interest was shown in the residue number system (RNS) as a basis for computational hardware. Back then, the RNS arithmetic was considered unsuitable for general purpose computers because of the complexity of the operations of division and comparison. Now more than ever, as digital signal processing (DSP) emerges as an important area within electrical engineering and VLSI fabrication and design techniques are developed, there is a renewed interest in the RNS. The RNS seems especially suitable for DSP because rapid computation using "simple" operations of addition/subtraction, and multiplication, is characteristic of DSP algorithms. The RNS also has the desirable properties for the VLSI implementation: modularity and fault tolerant capabilities. The authors believe that now it is time when the RNS takes a new direction towards multipurpose processing rather than to be locked in the realm of special purpose applications.

RNS is defined by a set of moduli m(i); i=1,2,...,N. These moduli are relatively prime pairwise positive integers. Based on the Chinese Remainder Theorem-CRT [32] there is a unique representation for each number in the range

$$0 \leq X < M, \text{ where } M = m(1) * m(2) *...* m(N)$$

Each integer X can be represented by a set of residues x(i) = [X] mod m(i). If it is desirable to deal with both positive and negative integers, the range is defined as:

$$[-(M-1)/2 \,;\, (M-1)/2\,] \text{ for M odd}$$

$$[-M/2 \,;\, (M/2)-1] \text{ for M even.}$$

Then each integer X can be represented by the N-tuple

$$x(i) = [X] \bmod m(i) \text{ if } X \geq 0,$$

$$x(i) = [\,M - |X|\,] \bmod m(i), \text{ if } X < 0.$$

In the RNS the operators # {+,-,*,:} are defined as follows:

$$\text{If } Z = X \# Y, \text{ then } z(i) = [\, x(i) \# y(i)\,] \bmod m(i).$$

Each residue digit can be computed independently of the others allowing fast data processing in N parallel independent channels or ALUs. There is evidence that residue arithmetic is theoretically the fastest way of performing addition, subtraction, multiplication, and polynomial transforms [31]. The main characteristic of this number system is that there are no carries between positions on separate moduli. A major advantage that the RNS has over conventional number systems is the relatively small size of the word lengths (not more than 4-6 bits) involved in each of the RNS "channels". This translates to a very small ALU per channel which means that in a hardware implementation, hundreds of ALUs may fit in a very small area. The properties of the RNS can be used to increase system performance essentially by introducing parallelism at the ALU level in addition to the parallelism at the architectural level. The arithmetic precision is easily modified without redesigning an old ALU and fault tolerance can be provided without too much additional complexity.

The above mentioned properties of RNS are extremely well matched to optical systems, especially if operations are represented as spatial reflections. For example, specific operations, such as addition by 3 or multiplication by 3, can be viewed as a mapping of the set of possible input residues (for a particular modulus) onto itself, but with a reassignment of values. Such a reassignment of residue values is called a map, which is illustrated in Fig. 1. Suppose that the modulus is 5, and it needs to represent, for all possible input residues, the result of adding a number with residue 3. As shown in Fig. 1(a), addition of 3 allows the number to be assigned to different inputs by a cyclic shift of 3 units. Fig. 1(b) shows the map that corresponds to multiplying any input residue by a number with residue 3 (again the modulus is 5). In this fashion, various arithmetic and more complex algebraic operations can be viewed as maps (for example, polynomial operations).

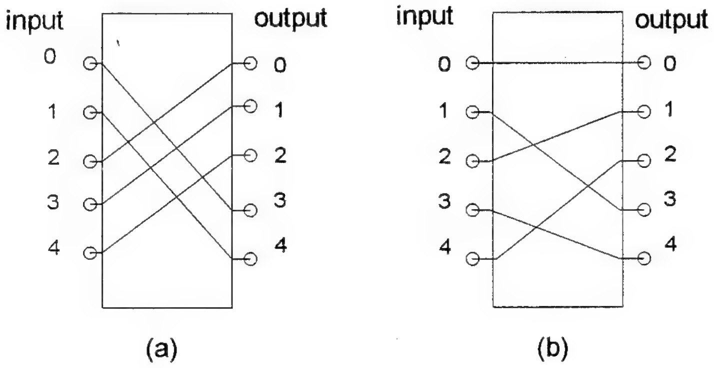

Fig. 1 Mappings of possible residues for modulus 5.
(a) shows addition by 3, notice the cyclic shift of 3 units.
(b) shows multiplication by 3.

An example of evaluation of polynomials by table lookup (map) for m(i) = 7 and $F(X) = X^2 - X + 1$ is shown in Table 1.

Table 1

X	$F(X) = X^2 - X + 1$
0	[00 - 0 + 1] mod7 = 1
1	[01 - 1 + 1] mod7 = 1
2	[04 - 2 + 1] mod7 = 3
3	[09 - 3 + 1] mod7 = 0
4	[16 - 4 + 1] mod7 = 6
5	[25 - 5 + 1] mod7 = 0
6	[36 - 6 + 1] mod7 = 3

The chosen representation of the physical states are spatial positions of a light beam. Hence the maps of interest must be capable of performing spatial permutations of the possible positions of a light beam. The examples considered correspond to a fixed operation which, in particular, always adds 3 to an incoming residue or multiplies 3 by a residue. A possible approach to constructing a changeable map is illustrated in Fig. 2. The modulus in this example is 5, so there are 5 input optical ports and 5 output optical ports. Suppose that a 10-bit number is to be added to the incoming optical residue. The map consists of 10 subunits, each subunit consisting of a planar waveguide with three switching channels, represented by three lines in the figure. A light beam incident at any point on a switching channel is assumed to be either transmitted or reflected, depending on the electrical signal applied to the switch. One of the three channels in each subunit is always along the diagonal of the subunit. If the binary number applied to that subunit is 0, the diagonal channel (labeled 0) is activated as the reflector, and no permutation of the ports occurs. If the binary number applied to the subunit is 1, the off-diagonal channel is transmissive, and the diagonal channels (labeled 1) become reflective. The positioning of the different weights (powers of 2) must be associated with each binary digit. The first subunit corresponds to the most significant bit. After reflection from the diagonal or off-diagonal channels, the reflected beam enters the second subunit, etc.

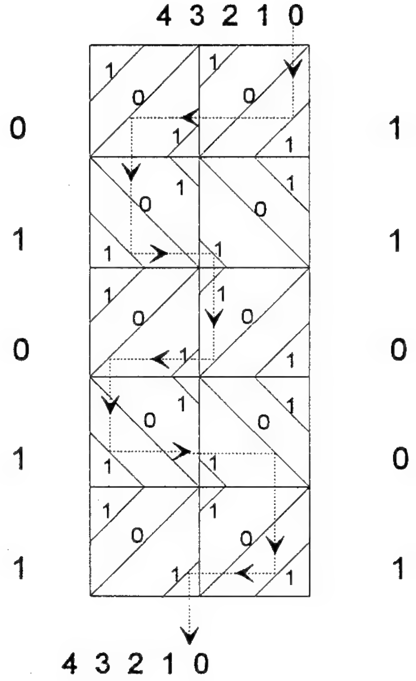

Fig. 2 Changeable map accomplished by integrated optic switching.

Optical systems are characterized by a very important property of the inherent parallelism. This property is the ability to have beams of lights crossing each other without any interaction. An extreme example of this parallelism is offered by a simple lens. By imaging a set of input ports onto a set of output ports, a reversal of the order of those ports can be accomplished. This reversal is reached without the necessity of providing separate electrical connections between inputs and outputs and without associated problems of connections crossing over connections. The integrated optic cyclic permutation device of Fig. 2 provides a second good example. With only three separate switching channels per subcell and only two electrical inputs to each subcell, a complicated cyclic permutation of many ports is accomplished. Methods that are based on optical 2D spatial light modulators (SLM), due to their parallel data processing capability, can achieve a high throughput. For a different optical logic and algebraic operations, using either binary or multiple-valued logic, a number of optical truth table processing schemes have been proposed [33,34]. The RNS is well suited for both content-addressable and location-addressable memory lookup processors.

The present Position-Coded RNS (PCRNS) lookup-table-processors utilize a 2D LED (laser diode) array[33], which is good only for small moduli. In order to eliminate this limitation, the operands can be represented as two mutually orthogonal light bars. Their intersection at an output point represents a particular result of an operation. For a specific operation, the collection of light output points constitutes a PCRNS lookup table.

The optical PCRNS lookup table evaluation is a two-stage process. First, a lookup table is generated and stored. Next, during a readout stage, based on indices of the operands x(i) and y(i), the operation's result is accessed. For example, in Fig. 3 a liquid crystal TVs (LCTV) may be divided into four parts, with each part representing a different modulus lookup table. Each multiplication operand is either a horizontal or a vertical light bar. With the light passing through the common image points, particular multiplication results are generated. For example, the lower right-hand corner point (3,5) represents the result of a multiplication by mod 7. The described optical processor requires, regardless of the RNS dynamic range, only a single light source.

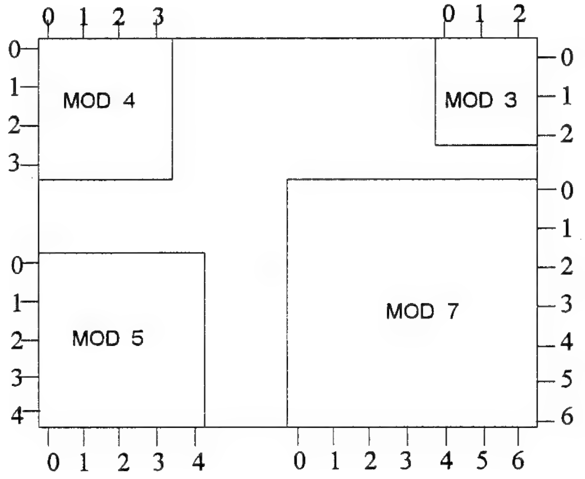

Fig. 3 LCTV screen partitions, where each partition represents a different modulus multiplication result.

An array logic may be also used as the building blocks of the optical processor. Much effort has been devoted to 2-D logic gate array devices and their materials [35,36], but the lack of a suitable optical equivalent to transistors still hinders the development of digital optical processors. The encoding concept for binary logic uses the two orthogonal polarization states of light and spatial separation of the modulated light [36]. The operation kernel, which determines the logic operation to be executed, simply filters the encoded light according to instructions. The input bit array is used as the address bits. Selected output bits are combined in a single bit that represents the value of the function[38]. The configuration of the array logic for two binary inputs is shown in Fig. 4.

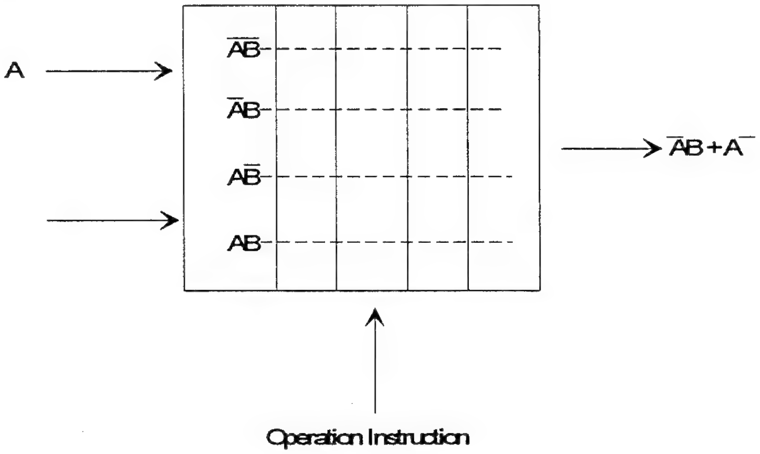

Fig. 4 Array Logic

Four logical products of inputs A and B are generated and arranged individually. Depending on the operation instruction, a combination of the products is selected as the output. Binary inputs are encoded, and the kernel determines the operation to be executed. Microchannel spatial light modulators [39] having photomultiplying capability are used as the polarization modulators for binary input patterns. A microcomputer-controlled liquid crystal spatial light modulator may be used as a real-time programmable operation kernel. As it was shown above, operations are carried out in cascade fashion. This cascading effect is an important aspect in building general purpose optical processors that can carry out operations having complex functional forms in a pipeline fashion. The usage of RNS can provide complex parallel and/or pipeline structures that in turn can supply not only high performance but also a high level of tolerance and good precision characteristics for different types of digital processing algorithms. The possibility of parallel processing of all positions in RNS allows performing algebraic procedures based on nomography by substituting two input tables for plane nomograms.

Nomography provides a full absence of carries (in comparison with methods of adding binary residues) and requires less information to be maintained (in comparison with a table method). Besides, the nomographic principle of accomplishing the RNS based on opto-electronics might provide an improvement in the level of performance from gigaflops to hundreds of gigaflops. It is possible to build simple nomograms based on leveled points. Examples of adding and multiplying residues on modulo 7 is shown in Fig. $5_{(a)}$ and $5_{(b)}$.

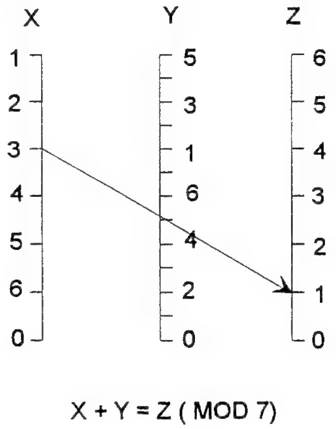

Fig. 5-a

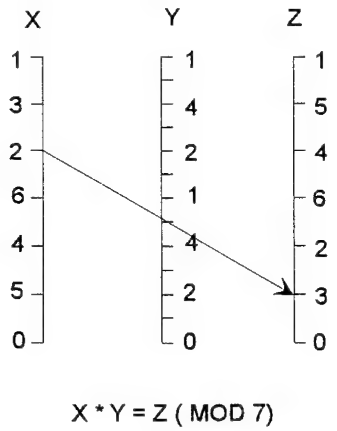

Fig. 5-b

Illustrated in Fig. 6 is an optoelectronic nomographic adder. Symbols 1 and 2 represent input registers for two residual operands. The electrical signals of the operands, corresponding to the residues of these operands, opens a cell (light-valve), marked as 6, in a waveguide, marked as 7 (based on a nomogram), and a corresponding light-valve in a parallel waveguide (8). Simultaneously a light beam passes through the waveguide (5) from the injector laser (4) and is divided into a set of light beams in these waveguides. The laser light beam, which passes through the opened light-valve element in the waveguide marked as 8 (that is through optical input and output), acts upon a corresponding photo diode in waveguide (9). The electrical signal that exits, corresponds to the sum of the two residues which is output into a register, marked as 3.

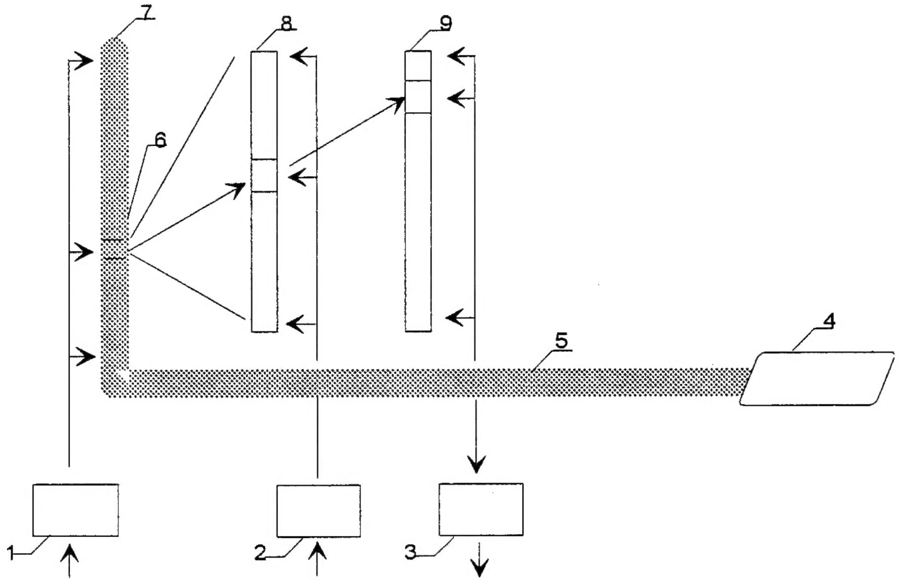

Fig. 6 Nomographic adder

By using composite and spatial nomograms, nomography allows performing several operations simultaneously and provides increasing speed to executing non-modulo procedures. An example of a composite nomogram for the function x + y + z = t on modulo 7 is shown in Fig. 7.

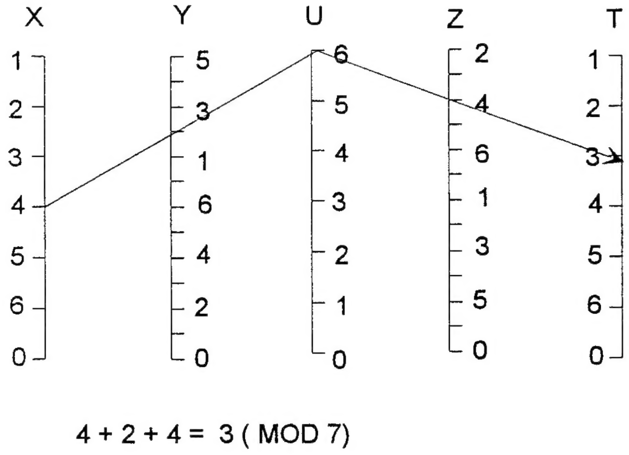

Fig. 7 Composite Nomogram

Current supercomputers have parallel and vector processing. The latter is found in specialized vector supercomputers like the Cray, in MPPs, in specialized vector processors included in the computer, like the IBM ES 9000. This class of computer techniques is intended for solving many different problems based on vector, vector-matrix, and matrix algebraic procedures as well as simulating complex systems (for example, the Monte-Carlo method), integer processing, and procedures of digital signal processing(DSP) such as convolution, correlation, digital filtering, and fast Fourier transforms (FFT). The potential of matrices and the RNS, however, is not in trivial matrix-to-matrix operations, but computing exact matrix inverses or solving systems of linear algebraic equations. The above mentioned procedures mainly consist of very "convenient" RNS operations such as multiplication, addition, and subtraction and often involves computing sum-of-products.

Thus, these procedures could be realized very effectively by vector processors (VP) using RNS and an optical technology. We hope that the optical RNS VP will possess essentially higher performance in comparison with an electronic VP. Creating such VP would allow us to discover the possibilities of realizing an optical general purpose RNS processor.

Even though the current optical and electronic devices require comparable activation times, the time required for propagation of a signal through the device may be much shorter in the optical case. Hence, if a long sequence of transferring or transforming data is required, the speed of the optical device may be faster. It becomes apparent in the electronic unit design process that major problems in propagation delay arise due to the difficulties associated with crossing electrical interconnections. Such difficulties can be avoided in some optical realizations, due to the ability of light beams to cross each other without any interaction.

5. Conclusion

In todays market, computer performance at the gigaflops level can only be provided by combining hundreds to thousands of processors as in a Massive Parallel Processor (MPP). Providing speeds significantly beyond teraflops will evidently require major innovations in computer technology. Optics technology may offer a breakthrough in performance, but it will require a rational rethinking of computer architecture and how the technology can support an appropriate parallel computational model. Recent advances in optical technology have led to a belief that optical computing offers good hope for solving problems for which electronics will be fundamentally limited. The well known advantages of the optical domain, such as huge bandwidth, very little interference, free-space distribution possibilities, and inherent parallelism are welcome as well.

Performance of MPP depends strongly on the network that interconnects the different collaborating processors. The availability of multiple parallel connections allows full connectivity, and is an essential prerequisite for an interprocessor network that is able to meet the ever growing communication demands.

The authors believe that only optics will be able to provide the high communication throughput required for a massive parallel MIMD machine with distributed shared memory system implemented with a single global address space.

The use of the residual number system (RNS) will be a key element in future processor speed-up. It's the opinion of the authors that the time is right to bring back the residual number system. There is evidence that residue arithmetic is theoretically the fastest way of performing algebraic procedures, including polynomial transforms, used very often in computing. The residue number system provides exceptional capabilities of discovering and correcting faults which are absent in conventional structures. It also possesses properties which are extremely well matched to optical systems, especially if operations are represented as spatial reflections.

The opinion reflected in this paper is that only through the combination of optics and the residual number system, will we be able to provide the breakthrough in performance necessary to reach up to the petaflops levels.

Nomography, incorporated with the residual number system, can even further increase processor performance. By using composite and spatial nomograms, nomography allows the execution of several operations simultaneously and provides increasing speeds of non-modulo procedures.

Thus optics in combination with the RNS allow us to realize a high performance and highly reliabled processor as the base for large scale multiprocessor systems with an effective optical interprocessor communication. The authors believe that this combination can provide a breakthrough in the performance of highly parallel computing.

6. Acknowledgments

This research was supported in part by NASA (under grant NAGW-4080),
ARO (under BMDO grant DAAH04-0024), NATO (under grant HTECH.LG-931449),
NSF (under grant CDA-9313624 for CATE Lab), and the State of Florida.

7. References

[1] P.C. Patton, "Multiprocessors: Architecture and Applications", IEEE Computer,Vol.16,June 1985,pp.29-40.

[2] W. Crowther et.al.,"Performance Measurements on a 128 Node Butterfly Parallel Processor",Proc. of the Int.conf. on Parallel Processing,1985,pp.531-535.

[3] M.A. Franklin, and S. Dhar, "On Designing Interconnection Networks for Multi-Processors",in Proceedings International Conference on Parallel Processing,Chicago (1986),pp.208-215.

[4] M.J. Quinn,"Designing Efficient Algorithms for Parallel Computers", Mc Graw-Hill,Inc.,1987.

[5] B. Furht, "Survey of Parallel Computers and Multiprocessors", Notes from the Advanced Summer Course on Architectures for VLSI Computers", L'Aquila,Italy,July 3-9,1988.

[6] P. Willey, "A Parallel Architecture Comes of Age at Last",IEEE Spectrum,Vol.24,No.6,June 1987,pp.46-50.

[7] L.N. Bhuyan,and D.P. Agrawal,"Generalized Hypercube and Hyperbus Structures for a Computer Network", IEEE Trans. on Computers, Vol.33,No.4,April,1984,pp.323-333.

[8] W.D. Hillis, "The Connect Machine (MIT Press,Cambridge,1985).

[9] L. Dekker, E.E. Frietman, "Optical Link in the Delft Parallel Processor", in Proceedings, Second European Simulation Multiconference,Nice (1988).

[10] J. Wilson, and J.F.B. Hawkes," Optoelectronics.An Introduction (Prentice-Hall International,London,1983).

[11] S.R. Forrest," Monolithic Optoelectronic Integration.A New Component Technology for Lightware Communication", IEEE Trans. Electron. Devices ED-32,26402655 (1985).

[12] S.H. Lce, S.C. Esener, M.A. Title,and T.T. Drabik," Two-Dimensional Silicon /PLZT Spatial Light Modulators:Design Considerations and Technology", Opt.Eng. 25,250-260 (1986).

[13] R.G. Smith and S.D. Personick,"Receiver Design for Optical Fiber Communication Systems",in Semiconductor Devices for Optical Communications, reviewed in Appl. Opt. (Springer-Verlag,Berlin,1980).

[14] D. Hartman," Digital High Speed Interconnections : a Study of the Optical Alternative", Opt. Eng. 25,1086-1102 (1986).

[15] J.K. Buttler, Semiconductor Injection Lazers (IEEE, New York, 1979).

[16] R.G. Walker," Broad band (6 GHz) GaAs/AlGaAs Electro-Optic Modulator with Low Drive Power",Appl. Phys. Lett.,54,1613-1615(1989).

[17] S. Wang and S. Lin," High-Speed III-IV Electro-Optic Waveguide Modulators at L=1.3 mm", IEEE/OSA J.Lightware Technol. LT-6,758-771 (1988).

[18] G. Vella-Coliero," Optimization of the Optical Sensivity of pin FET Receivers", IEEE Electron Device Lett. EDL-9,269-271(1988).

[19] K.-H. Brenner, and Huang," Optical Implementations of Perfect Shuffle Interconnection", Appl. Opt. 27,135-137 (1988).

[20] T. Mineniot, S. Numata, and K. Miyamoto," Optical Parallel Logic Gate Using Light Modulators with the Pockets Effect: Applications to Fundamental Components for Optical Digital Computing", Appl.Opt. 25,4046-4052 (1986).

[21] K. Iga, Y. Kokubun, M. Oikawa," Fundamentals of Microoptics", (Academic,New York,1984).

[22] E. Bradley, and P. Yu," System Issues Relating to Lazer Diode Requirements for VLSI Holographic Optical Interconnects", Opt. Eng.28,201-211 (1989).

[23] L.A. Bergman,W.Wu,A. Johnston,and R.Nixon,"Holographic Optical Interconnects for VLSI", Opt. Eng. 25,1109-1118 (1986).

[24] A.R. Johnston,L.A. Bergman,and W. Wu," Optical Interconnection Techniques for Hypercube", Proc. Soc. Photo-Opt. Instrum. Eng. 881, 186-191 (1988).

[25] C.T. Sullivan, and A. Husain," Guid-Wave Optical Interconnects for VLSI Systems", Proc. Soc. Photo-Opt. Instrum. Eng. 881, 27-00 (1988).

[26] R. Selvaraj, H. Lin,and J. McDonald," Integrated Optical Waveguides in Polyimide for Wafer-Scale Integration", IEEE/OSA J.Lightwave Technol. LT-6,1034-1044 (1988).

[27] C. Harder, B. Zeghbroeck,H. Meier,W. Patrick, and P. Zettiger," 5.2 GHz Bandwidth Monolithic GaAs Optoelectronic Receiver", IEEE Electron Devices Lett. EDL-9,171-173 (1988).

[28] C.P. Kruscal and M. Smir," The Importance of Being Square", in Proc.,Eleventh Int. Symp. on Comp. Archit. (1984), pp.91-98.

[29] Snyder,L., " Type architecture,shared memory and the corollary of modest potential",Annual Rev. Comput.Sci. 1 (1986),289-318.

[30] J. Giglmayr, "Compact interconnection networks for photonics", Appl. Opt.,Vol.32,No.26,September,1993,pp.5002-5009.

[31] W.K. Jenkins, and F.J. Lcon, "The use of residue number system in the design of finite impulse response filters", IEEE Trans. Circuits Systems,Vol. CAS-24,Apr. 1977.

[32] N.S. Szabo and R.I. Tanaka, Residue Arithmetic and Its Applications to Computer Technology (McGraw-Hill,New York,1967).

[33] A.P. Goutzoulis, D.K. Davis,and E.C. Malarkey, "Prototype Position-Coded Look-up Table Using Laser Diodes", Opt. Commun. 61, 302 (1987).

[34] M.M. Mirsalehi and T.K. Gaylord, "Truth-Table Look-up Parallel Data Processing Using Content-Addressable Memory " , Appl. Opt. 25, 2277 (1986).

[35] G. Livescu,D.A.B. Miller,J.E. Henry,A.C. Gossard,and J.H. English, "Spatial Light Modulator and Optical Dynamic Memory Using 6*6 Arrays of Self-Optic-Effect Devices", Opt. lett. 13,297-299 (1988).

[36] J. Tanida, J. Nakagawa, and Y. Ichioka, "Birefringent Encoding and Multichannel Reflected Correlator for Optical Array Logic", Appl.Opt. 27, 3819-3823 (1988).

[37] T. Kurokawa and S. Fukushima, "Real Time Image Processing Based on Optical Array Logic", in Optical Computing 88, Eg, Toulon, France (Aug., 1988).

[38] H. Fleisher and L.I. Maissel, "An Introduction to Array Logic", IBM J. Res. Dev.19, 98-109 (1975).

[39] T. Hara, Y. Kato,and Y. Suzuki, "Microchannel Spatial Light Modulator with Improved Resolution and Contrast Ratio",Proc. Soc. Photo-Opt. Instrum. Eng. 613, 153-157 (1986).

[40] G. Almasi, and A. Gottlieb, Highly Parallel Computing (Benjamin/Cummings, New York, 1989)

[41] A. Tanenbaum, Structure Computer Organization (Prentice-Hall, 1990)